Mark as done

Mini Data Platform

Goal

Build a basic data platform using Docker Compose that can:

1. Collect data
2. Process data
3. Store data
4. Visualize data

4 Main Components (All in Docker)

1. **Database:** PostgreSQL - stores processed data
2. **Processing:** Apache Airflow - runs data pipelines
3. **Storage:** MinIO - file storage (like AWS S3)
4. **Dashboards:** Metabase - creates charts and reports

Simple Project Flow

Part 1: Set up all 4 services with Docker Compose

Part 2: Create a data pipeline that processes CSV files

Part 3: Build dashboards to show the data

What to Build

1. **Docker Compose file** - runs all services together
2. **Sample data generator** - creates fake sales/user data

3. **Data pipeline** - cleans and processes the data
4. **Dashboard** - shows charts and graphs

Example Use Case

Sales Data Platform:

- Upload sales CSV files to MinIO
- Airflow processes the files and loads to PostgreSQL
- Metabase shows sales charts and trends

Feel free to use any use case of your choice

Git Repository Submission Requirements

Repository Setup:

- Create a GitHub repository for your team
- Include a comprehensive **README.md** with setup instructions
- Add proper **.gitignore** for Docker volumes and logs
- Organize code in clear folder structure

Required Documentation:

- Setup instructions in **README.md**
- Architecture diagram showing data flow
- Screenshots of working dashboards
- Team member contributions (If working in a team)

Assessment (Simple)

- Does it run? (440%)
- Does data flow through all components? (40%)
- Are the dashboards useful? (20%)

Create a GitHub Actions pipeline that automates the following:

- **CI:** Build and test Docker images for each service on every commit.

- **CD:** Deploy updated containers to a test environment automatically.
- **Data Flow Validation:** Run automated checks ensuring data moves successfully from ingestion (MinIO) → processing (Airflow) → storage (PostgreSQL) → visualization (Metabase).

Last modified: Monday, 12 January 2026, 10:33 AM